

-2-

The present invention is directed to a system having a pointer into a segment of memory where the system must be able to determine from the pointer the full bounds of the memory segment. The invention has particular application to pointers which include a permission field which indicates how a process may access data within the segment. As illustrated in Fig. 5, the segment comprises multiple blocks and the address may point to any location in any one of those blocks. As discussed at page 13 with respect to Fig. 4, the base address of a block can readily be determined by setting the B least significant bits equal to zero. However, that approach does not allow the base address of the full segment to be determined. In accordance with the invention of claims 1-36, a finger field in the pointer denotes the block of the segment into which the address points. That information along with the block size and number of blocks in a segment defined by the block and length fields, allows ready computation of the full bounds of the segment. In that way, bounds checking can be performed without using additional memory references to segment bounds.

Wooten discloses a system for enabling a processor to use different addressing modes and map memory instructions in those modes onto physical memory. As used in Wooten, the term "segment" refers to an independent address space (column 9, lines 28-29). "An address issued by a program is a logical address and consists of a selector and an offset. Generally, the selector identifies a particular segment, and the offset identifies an address within the segment." (Column 9, lines 35-45). Selector values are loaded into segment registers (Fig. 4, reference 402).

The Applicants disagree with the Examiner's assertion that Wooten discloses an address pointing into a segment of memory and a finger field which denotes a block of the segment of memory into which the address points, as recited, for example, in claim 1. The Examiner states that the address pointing into the segment of memory is shown in Wooten as reference 402 in Fig. 4. However, reference 402, Fig. 4, of Wooten shows a segment base address. The segment base address always points *to the beginning* of the segment, not *into* the segment of memory, as claimed in claim 1 of the present application.

In discussing the finger field, the Examiner suggests that the offsets taught in Wooten are analogous to the finger field. The Applicants respectfully disagree. In Wooten, offsets are used

-3-

to refer to addresses in memory; such reference is sufficient to describe a precise location in memory because the base address of the current section of memory being addressed is located in the segment register (column 9, lines 35-37). As quoted from Wooten above, *both* the base address of the segment and the offset are needed to indicate an address pointing to a location in memory; together they are analogous to an address pointing into the segment of memory. As claimed, the finger field is distinct from the address.

Wooten discloses only a way to address a location in memory and does not teach a finger field which denotes a block into which the address points, as recited in claims 1, 10, 19, and 28, which are therefore not anticipated by Wooten. Dependent claims 2-9, 11-18, 20-27, and 29-36 depend on the corresponding independent claims and, therefore, are also not anticipated by Wooten. Reconsideration is respectfully requested.

In connection with dependent claims 2, 11, 20 and 29, the Examiner states that Wooten teaches the pointer to a location of memory further comprising a permission field which indicates *how* a process may access data within the segment of memory (Fig. 4, reference 406); however, the permissions of Wooten are *whether* a process may access data within a particular segment of memory, not *how* it may access it (column 10, lines 10-25). Therefore, claims 2, 11, 20, 29 and other dependent claims, are further not anticipated by Wooten.

Claims 37-40 are directed to front-padded allocation discussed at pages 15-16. As discussed, these claims have application to languages such as Java where a pointer is always to the first word of an object. Front-padded allocation means that the entire region of memory in which an object is stored is in the end portion of the memory segment. Thus, where an object of size N is allocated into a segment of size M, where M is greater than N, the first word of the object is located at word M minus N of the segment. In that case, the finger field is not required because it is known that the pointer will always point somewhere into the first block of an object.

Although Wooten identifies a base address of a segment, it is not the base address of a memory region within the segment, where all portions of the memory segment not within the

-4-

memory region have addresses less than the address in the pointer. That is, Wooten does not suggest front-padded allocation. Accordingly, claims 37-40 should be allowed.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned at (978) 341-0036.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By 

James M. Smith

Registration No. 28,043

Telephone: (978) 341-0036

Facsimile: (978) 341-0136

Concord, MA 01742-9133

Dated: 